

SAHIB BHARAJ

FULL-STACK ENGINEER — NODE · TYPESCRIPT · REACT

Nairobi, Kenya · 0xbharaj@gmail.com · bharaj.sbs · linkedin.com/in/sahib-bharaj · [GitHub 0xbharaj](https://github.com/0xbharaj) · [sirheeb](#) · [s13bp](#)

CORE STACK

Backend — Node.js · Express · TypeScript · Python · REST API design · webhooks · cron jobs

Data — MongoDB · PostgreSQL · Mongoose · Drizzle · Supabase · Redis

Frontend — React · Next.js · TypeScript · Tailwind CSS · Radix UI

Payments — Stripe · HMAC verification · reconciliation · ledgers · settlement · Solana SPL

Real-time & AI — Socket.IO · WebSockets · Model Context Protocol · Vercel AI SDK

HOW I BUILD

- **Never trust the client with an amount.** Totals recomputed from stored lines; the browser's word for "paid" is not evidence.
- **Verify, then reconcile.** Signatures checked constant-time over raw bytes — then reconciled anyway, because webhooks go missing.
- **State moves one way**, so a repair racing a callback cannot un-pay somebody.
- **Configuration over deploys** for rates, limits and commercial terms.

LEADERSHIP

- Led **15+ developers** on applied computer vision at ManyMangoes.
- Lead a small engineering team across systems and dev-ops as **CTO at ZKOSIS**.
- Work directly with **compliance partners and security auditors** on infrastructure built to be verified by people paid to find the hole in it.

PROFILE

In 2022 I led a team of 15+ developers building image recognition for mangrove seedlings — counting and classifying them from survey imagery, back when telling people you did AI for a living got you a raised eyebrow. That project set the pattern for everything since: a hard technical problem, a team to point at it, and a result somebody had to be able to trust.

Today I am CTO at ZKOSIS, leading a small engineering team while architecting a zero-knowledge identity attestation engine. Hands-on across the stack — typed Node services, the schemas beneath them, the React interfaces on top — and most of it involves money: payment links with verified webhooks and reconciliation, custodial ledgers, scheduled payout engines.

EXPERIENCE

Chief Technology Officer ZKOSIS

May 2025 – Present

Remote

- Architect and develop a **ZK-based identity attestation engine** for compliant, anonymous identity verification on-chain.
- **Lead a small, high-performance engineering team** across systems and dev-ops.
- Collaborate with compliance partners, security auditors and DeFi protocols to integrate zkosis as a trust layer.

Chief Technology Officer OxCapital Finance

2024 – Present

Dubai, UAE (remote)

- Technology leadership and project finance for a capital-management platform.

Frontend Developer iii6

Dec 2022 – Present

Germany (remote)

- Production front ends and Node services, delivered remotely into a European team.

Advanced Technology Manager ManyMangoes

May – Oct 2022

Remote · Applied computer vision

- **Led a team of 15+ developers** delivering an image-recognition system for **mangrove seedlings** — detection and classification over survey imagery, on a YOLOv4 pipeline.

Dapp Engineer SafeConnectX

Oct – Dec 2022

Atlanta, Georgia (remote)

- Node and React application work over on-chain contracts, following an earlier engagement as **Blockchain Developer** (Oct 2021 – May 2022) in React and Solidity.

SELECTED WORK

uXRP — Solana ↔ XRPL bridge

2025

[u-xrp-front.vercel.app](#) · TypeScript · Express · MongoDB · Redis · xrp1

- Transfers move as a **worker pipeline** rather than on a request thread — deposit watcher, confirmation processor, business-logic stage, sender — with exchange rates and fee configuration updated out of band behind a validator.
- **Built around the failure modes.** XRPL ledger-sequence expiry retried automatically with a fresh sequence, reserve-balance alerting below threshold, stuck-transaction detection, and an admin console that can retry a payment or pause the whole system.

QubiNodes — DeFi yield platform

2026

TypeScript · Express · MongoDB · ethers.js · node-cron

- Earnings accrued per position, distribution resolved on a cycle, and settlement run as a **scheduled payout engine** — with auto-claim, a referral tree and vault reconciliation.

Yasmin's Kitchen — live commerce and reconciliation

2026

[yasminskitchen.com](#) · Next.js · Stripe · MongoDB

- **Payment links** restricted to one session so a forwarded URL cannot be paid twice, amounts recomputed server-side in integer pence, webhooks verified by hand, and a reconciler for the callback that never lands.